# 07 | 零配置组网：设备如何发现彼此?

2020-11-23 郭朝斌

物联网开发实战 进入课程 >

**讲述：郭朝斌**
时长 17:32 大小 16.07M

你好，我是郭朝斌。

不知道你还记不记得，在基础篇的第 2 讲中，我介绍了 Wi-Fi 设备的配网方式，比如一键配网技术（Smart Config）和设备热点配网技术。这些技术已经可以实现**一定程度的自动化**，让设备比较方便地连接上 Wi-Fi 热点。

同时，我也提到了零配置配网方式，它试图通过已连接上 Wi-Fi 热点的设备，来实现可信任设备**完全自动化**的配网。

但是，你可不要把**零配置配网**（Zero Configuration Provisioning）和我们这一讲要谈的**零配置组网**（Zero Configuration Networking，ZEROCONF）混淆了。配网只是第一

步，因为物联网设备无法方便地输入信息，所以相比手机或电脑，我们引入了这个额外的操作步骤。

配网成功之后，我们还需要组网，也就是让设备获得一个**自己的 IP 地址**，同时也知道局域网内的**路由器**（Router）的 IP 地址和 **DNS**（Domain Name System）的 IP 地址等信息。设备自己的 IP 地址是它在 TCP/IP 网络中的唯一标识；路由器可以把设备的数据包正确地转发出去；而 DNS 服务器可以帮忙解析出数据包中需要设置的目的地 IP 地址。

> 温馨提示：如果你对 IP 地址和 DNS 这样的基本概念还不是很熟，可以学习《趣谈网络协议》等课程临时恶补一下。因为这一讲还会涉及不少网络基础知识，对它们有一定了解的话，你的学习效果会更好。

零配置组网就是把这些工作自动化，达到不需要用户手动去操作，甚至实现这个设备可以与网络内的其他设备配合工作的目的。

比如，你要为自己的智能家居系统加一个投屏设备（它可以外接普通电视或者显示器，让你可以将手机或者电脑的屏幕在大屏上显示出来）。你希望这个投屏设备接入 Wi-Fi 热点之后，你就可以在手机上直接发现它，并且马上开始投屏，看自己想看的电影。

这个想法非常美，但是你知道要怎么实现吗？其实并不难，你只要掌握我在这一讲介绍的零配置组网就可以了。

## 设备如何获取 IP 地址？

零配置组网的第一步，就是自动分配 IP 地址。

不知道你以前有没有这样的经验，在学校宿舍或者公司，当你的电脑通过网线连接网络时（现在用网线连网的方式已经不那么常见了），有时电脑桌面右下角会弹出气泡，提示“IP 地址冲突”。

这是因为你的电脑的 IP 地址，被网络内其他电脑占用了。所以你就需要打开网络配置的界面，手动填写一个不同的 IP 地址。对于非专业人士来说，这并不是一个很简单的事情；而且对于特殊的**子网掩码**（比如不像“255.255.25**5**.0”这样整齐，而是“255.255.25**3**.0”这种），即使有经验的用户，可能也很难确定**子网号**和**主机号**分别是什么。

现在随着 Wi-Fi 的普及，我们连网的时候基本上不会再碰到这种情况了。这一方面是因为每一个 Wi-Fi 热点接入的设备数量是有限的，另一方面则是因为 Wi-Fi 路由器提供了完善的 **IP 地址自动分配功能**。

## DHCP 协议

这个自动分配功能是基于 **DHCP 协议**（Dynamic Host Configuration Protocol，动态主机配置协议）实现的。DHCP 协议的前身是 **BOOTP 协议**（Bootstrap Protocol），从“Bootstrap”这个名字你就可以看出来，它解决的正是一个设备接入 IP 网络后，需要完成的第一件事——获取 IP 地址。

在网络发展的初期，接入网络的设备很少，所以，网络地址的分配和设置都是管理员手动来完成的。你可能想象不到，最初整个互联网的 DNS 数据库都是手动维护和更新的。

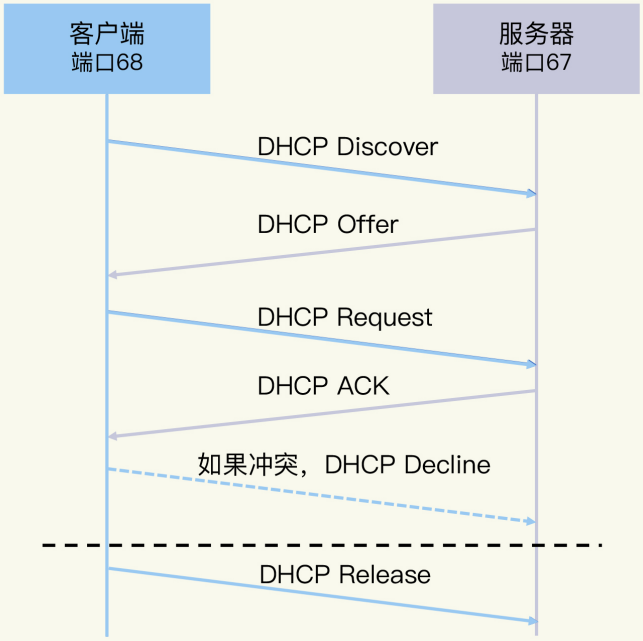
但是随着接入网络的设备越来越多，而且设备更加便携，经常移动来移动去，手动的方式就跟不上时代发展了。DHCP 协议正是在这种背景下出现的。

也许你对 DHCP 的名字很熟悉，但我还是想在这里稍微展开一下，带你了解它背后的工作原理，这样你就能在开发中更好地使用它。

DHCP 使用了服务器 - 客户端的架构模型。

- 当一个设备（你的手机）接入网络时，它自己就会作为 DHCP 客户端，请求网络地址。
- 然后 DHCP 服务器（家里的 Wi-Fi 路由器）会从地址池中挑选一个 IP 地址，分配给这个设备。
- 当设备不再使用这个 IP 时（你带着手机出门 / 睡觉飞行模式），DHCP 服务器会进行回收，之后再分配给其他有需要的设备（你新买的平板）使用。

DHCP 服务器与设备之间的通信是通过 **UDP 传输协议**完成的。因为 UDP 有一个优势，那就是不需要提前建立连接关系。DHCP 服务器的端口号是 67，设备的端口号是 68，它们一般的交互过程是这样的：




1. **Discover**：设备以广播的方式发送 DHCP Discover 消息，表示需要获取 IP 地址。
2. **Offer**：DHCP 服务器收到这个消息后，会发出 DHCP Offer 消息，作为回应。消息中带有 DHCP 服务器为设备分配的 IP 地址，也会包含其自身的 IP 地址。
3. **Request**：设备收到 DHCP Offer 消息后，将会广播一条 DHCP Request 消息，正式向 DHCP 服务器请求这个 IP 地址。
4. **ACK**：DHCP 服务器收到 DHCP Request 消息后，会判断服务器 IP 是否和自己的地址一致。如果一致，马上向设备回复 DHCP ACK 消息，并指定好 IP 地址的租用期限。
5. **Decline**：设备收到 DHCP ACK 消息后，还会验证一下 IP 地址是否可用。如果地址冲突，就说明不可用，它会发出 DHCP Decline 消息；如果地址不冲突，就可用的，设备将会按照租期使用这个 IP 地址。
6. **Release**：当设备不使用这个 IP 地址时，设备可以通过发送 DHCP Release 消息，来释放它。这样 DHCP 服务器可以重新分配这个 IP 地址。

## 如何让手机自动发现投影仪呢?

借助 DHCP 协议，当你的投屏设备接入 Wi-Fi 热点后，就可以自动获得一个自己的 IP 地址，比如“192.168.1.100”，同时还会自动得到路由器（Router）的 IP 地址，并且完成 DNS 的 IP 地址的自动设置。

那接下来，怎么让手机自动发现它，然后直接使用呢?

不知道你留意过没有，如果你新买了一台打印机，当这台打印机连上家里的 Wi-Fi 之后，你在电脑上打印文件时，它会自动显示出来供你选择。这是怎么实现的呢？

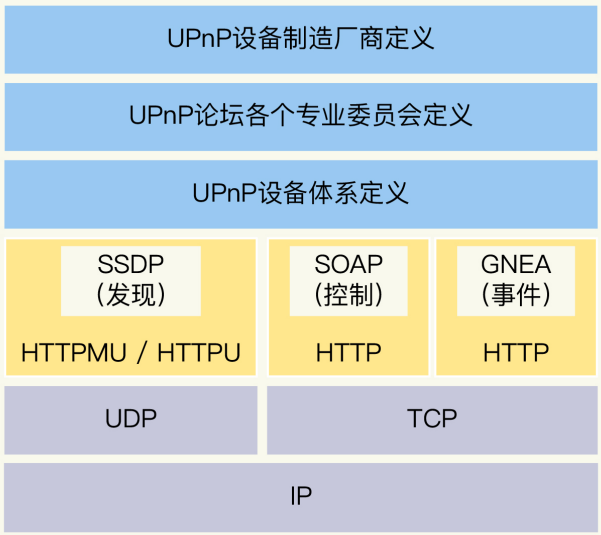
分析一下，一定是电脑通过某种方式知道有这台设备，并且知道它就是打印机，可以提供打印服务。打印机被电脑自动识别的这个过程，就是借助 UPnP 协议完成的。

## UPnP 协议

UPnP 是 Universal Plug and Play 的简称，它要实现的目标就是网络设备的即插即用。

UPnP 由设备寻址、设备发现、设备描述、设备控制、事件通知和基于 HTML 的描述界面六部分构成。其中设备寻址同样是基于我刚介绍过的 DHCP 实现，如果网络内没有 DHCP 服务器，UPnP 会基于自己的 AutoIP 方法指定一个 IP 地址。

从整体看，UPnP 是一个**多层协议构成的框架体系**，每一层都以相邻的下层为基础，同时又是相邻上层的基础，直至达到应用层为止。你可以参考下面的图片。




这里我重点介绍一下第三层（从下往上数），它基于 HTTP、HTTPU、HTTPMU 协议，属于传送协议层。传送的内容都是经过“封装”之后，存放在特定的 XML 文件中的。用于**设备和服务发现**的 SSDP（Simple Service Discovery Protocol，简单服务发现协议）协议就是基于 XML 传送数据的。

你不要被 SSDP 协议的名字迷惑了，其实它既提供了服务发现的功能，也提供了设备发现的功能。我们可以基于 SSDP 中的 M-SEARCH 方法来查询设备，然后基于设备的响应，

获得设备的服务能力的描述信息。同时设备可以通过 NOTIFY 方法向网络通知自己的服务能力。

借助 UPnP 协议，你的设备就可以自动被发现和使用了。这个自动化的过程，我们通常用一个专有名词来概括，也就是零配置组网。梳理一下，零配置组网包括三个方面的技术内核：

1. 为网络设备自动分配 IP 地址，一般涉及 DHCP 协议和 AutoIP 方法；
2. 自动发现和解析设备，主要是基于 SSDP 协议；
3. 自动传播和发现各网络设备提供的服务，主要也是基于 SSDP 协议。

这里我贴一下我家里小米电视盒子的一个 UPnP 的组播包的内容。它每 3 秒就有一包，这样用户体验更好，手机或者电脑投屏时更容易发现它。

复制代码

```
NOTIFY * HTTP/1.1
HOST: 239.255.255.250:1900
CACHE-CONTROL: max-age=66
LOCATION: http://192.168.31.188:49152/description.xml
NT: urn:schemas-upnp-org:service:ConnectionManager:1
NTS: ssdp:alive
SERVER: Linux/3.14.29, UPnP/1.0, Portable SDK for UPnP devices/1.6.13
USN: uuid:F7CA5454-3F48-4390-8009-483842e84c17::urn:schemas-upnp-org:service:C
```

你可以发现，它是基于 HTTP1.1 版本，准确地说，是 HTTPMU 协议；同时你可以看到组播地址，端口号是 1900。

那为了实现 UPnP 协议，你可以使用哪些开源的代码实现呢？

比较流行的开源库是 libupnp，小米电视盒子使用的就是这个开源库。另外还有一个选择是 GUPnP 项目，它包括几个不同的子项目，比如实现 SSDP 协议的 GSSDP 项目。

Android 平台上有一个开源库 Cling，它是 4th Line 组织开发的，你可以参考。现在也只能说“参考”，使用的话，需要谨慎评估，因为它已经不再被维护。

## mDNS 和 DNS-SD

除了 UPnP 协议，零配置组网还可以使用别的协议标准，比如 mDNS 和 DNS-SD 协议。

说起这两个协议，你可能比较陌生，不过苹果设备的 **AirDrop 功能**你可能比较熟悉。AirDrop 是通过 **Bonjour 服务**来发现网络上的其他苹果设备的。这个 Bonjour 服务就是 mDNS 协议和 DNS-SD 协议的具体实现。

**mDNS**（Multicast DNS）协议允许设备在本地的 DNS 名字空间，设置一个本地的域名。之后被询问的时候，它就通过 UDP 把 IP 地址广播出来，这样其它的设备就可以找到它。你可以简单地把 mDNS 理解为 **DNS 的本地网络版本**。

比如，主机 A 是 FTP 服务，它接入网络，并开启了 mDNS 服务，就会向 mDNS 服务注册服务信息：

我提供 FTP 服务，我的 IP 是 192.168.1.101，端口是 21。

当主机 B 接入相同的网络时，如果它向主机 B 的 mDNS 服务请求，需要找局域网内 FTP 服务器，主机 B 的 mDNS 就会在局域网内向其他设备的 mDNS 询问。然后，它就会找到主机 A 的 IP 地址和端口号。

**DNS-SD**（DNS Service Discovery） 协议，一般是和 mDNS 一起使用的。它使用三种 DNS 协议的记录类型来定义协议内容，三个记录分别是：PTR 记录、SRV 记录和 TXT 记录。它提供了服务发现的功能，作用类似于上面讲到的 SSDP 协议。

## 小结

总结一下，在这一讲中，我介绍了零配置组网的相关技术。零配置组网是一个技术组合，它要实现的目的是自动化网络设备的初始配置过程，让用户不需要额外的手动操作。

1. 零配置组网的第一步是为网络设备自动分配 IP 地址。DHCP 协议是实现设备寻址的标准技术。它是工作在 UDP 协议之上的应用层协议，提供了完善的 IP 地址请求、IP 地址分配和租期管理等功能。如果网络内没有 DHCP 服务器，在 UPnP 中定义了 AutoIP 方法，设备可以自己分配 IP 地址。在最新的 UPnP 规范中 AutoIP 切换到了 Link-Local Addressing 规范。这类 IP 地址一般是“**169.254**.0.0/16”范围的地址。

2. 第二步是自动发布和解析设备的名称。在 UPnP 中一般是基于 SSDP 协议，另一种替代的方案是 mDNS 协议。
3. 第三步是自动传播和发现各网络设备提供的服务，一种选择是采用 SSDP 协议；另一种方案是基于 mDNS 的 DNS-SD 协议。

UPnP 协议整合了 DHCP、AutoIP 和 SSDP 等协议规范，提供了一个完整的组网协议框架。UPnP 协议实现的功能非常完善，比如，基于 SOAP 提供了设备控制的能力，但是也引来了很多安全隐患，这个在后面我还会介绍到。

我在这一讲介绍的方法主要关于 Wi-Fi 设备的。你可能想问，那蓝牙和 ZigBee 设备怎么办呢？其实关于蓝牙和 ZigBee 这类设备，智能家居厂家一般都会制定自己的私有协议，比如小米有 **Mibeacon 协议**，我会在实战篇再为你介绍。

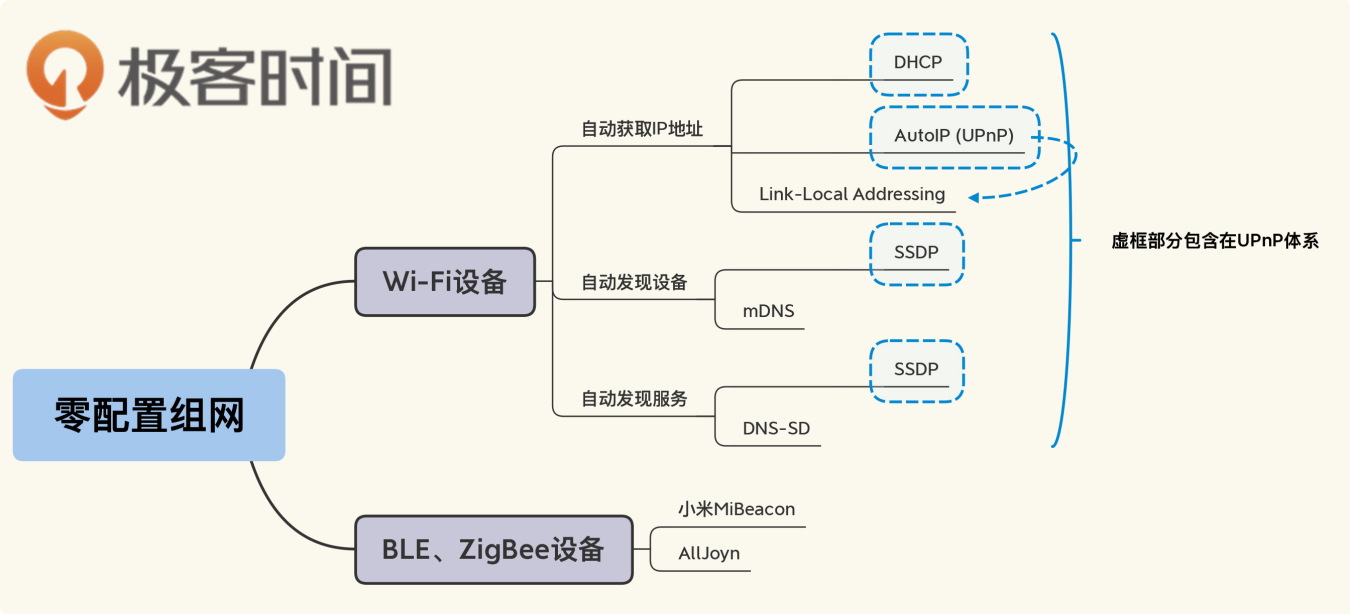
除了这些私有协议，行业内也有一些标准组织主导的开放协议，比如 **AllJoyn 协议**。它为蓝牙和 ZigBee 设备提供了设备发现和控制的解决方案。现在，AllJoyn 已经和 IoTivity 合并。这个合并是为了在物联网场景下，让零配置组网技术进一步发展。

具体来说，因为物联网设备用了很多不同的通信技术，所以 AllJoyn 提供了一个抽象层。它为底层网络协议栈定义了统一的接口，使得软件工程师可以相对容易地添加和安装新的网络。

AllJoyn 还采用了一种易于理解的对象模型和远程方法调用（RMI）机制，并且它重新实现了总线协议，基于 D-BUS 规范和扩展 D-BUS 协议，用来支持分布式设备。

因为相对来说，AllJoyn 协议的行业应用还处于早期，我就不展开介绍了，有这方面需求或者兴趣的话，推荐你可以到它的官网详细了解。

我这里总结了一个思维导图，供你参考。




## 思考题

最后，我还是给你留一个思考题吧。

在 DHCP 协议的流程介绍中，我提到当设备收到 DHCP 服务器响应的 DHCP ACK 消息后，并不会立即使用这个 IP 地址，而是要先检测一下 IP 地址是否有效。那设备是通过什么方式检测 IP 地址的有效性的呢？

你可以在留言区和我交流你的思考。同时，也欢迎你将这一讲分享给你的朋友一起交流学习。

提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。 页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 06 | 物模型：如何定义智能电灯？

下一篇 08 | MQTT：在实践中掌握一个通信协议

## 精选留言 (5)

写留言

**rmliu**
2020-11-23

思考题答案：
设备在收到DHCP服务器的 DHCP ACK报文后，广播发送 ARP报文，探测是否有其他的设备使用该IP地；
1）如果收到回应，就说明地址冲突，发送 DHCP Decline 报文到DHCP服务器，重新请求 IP;...
展开

作者回复: ❀❀❀❀❀❀完全正确

6 9

**sundy**
2020-11-27

老师 还有一个问题，发现设备为什么不采用udp广播的形式去发现?

作者回复: 是指广播和组播的选择吗？组播可以通过路由器转发，广播只能在局域网呢。

1



**阿冲**
2020-11-23

老师你好，配网成功后不就能获取到IP地址了，为什么要等组网阶段才能获取。这个有点疑问。

作者回复: 配网只是知道wi-fi热点的名称和密码，IP地址后面根据DHCP协议获取的。只是这个过程是自动的，所以用户没有感知。你可以进入管理界面，关闭wi-fi路由器的DHCP功能试一试，就了解它的作用了。

1

**sundy**
2020-11-27

老师 请问android上使用UPnP 怎么使用

展开

作者回复: Cling。课程中有介绍到。使用前评估一下。

**一步**

2020-11-25

利用DHCP 协议分配 ip 地址的时候， 客户端需要和服务端交互两次 Offer & ACK ，是为了防止别人冒充DHCP 服务器吗？ 要不然 当接收到 服务端的 Offer 应答的时候，就可以使用 ARP 协议广播来验证IP是否可用了

展开

作者回复: 防止不了冒充，不过网络中确实有可能有多个dhcp服务器，又是广播，所以需要确定从哪个DHCP请求。

1